

(19)日本国特許庁 (J P)

(12) 公 開 特 許 公 報 (A)

(11)特許出願公開番号

特開2000-348111

(P2000-348111A)

(43)公開日 平成12年12月15日(2000.12.15)

(51)Int.Cl. ⁷	識別記号	F I	テマコード*(参考)
G 0 6 F 19/00		G 0 6 F 15/22	N 5 B 0 4 9
9/46	3 6 0	9/46	3 6 0 B 5 B 0 8 2
12/00	5 1 8	12/00	5 1 8 A 5 B 0 9 8
17/60		15/21	Z

審査請求 未請求 請求項の数5 O L (全 25 頁)

(21)出願番号 特願平11-153330

(22)出願日 平成11年6月1日(1999.6.1)

(71)出願人 000005108

株式会社日立製作所

東京都千代田区神田駿河台四丁目6番地

(72)発明者 紅山 伸夫

神奈川県川崎市幸区鹿島田890番地 株式
会社日立製作所システム開発本部内

(72)発明者 溝手 裕二

神奈川県川崎市幸区鹿島田890番地 株式
会社日立製作所システム開発本部内

(74)代理人 100083552

弁理士 秋田 収喜

最終頁に続く

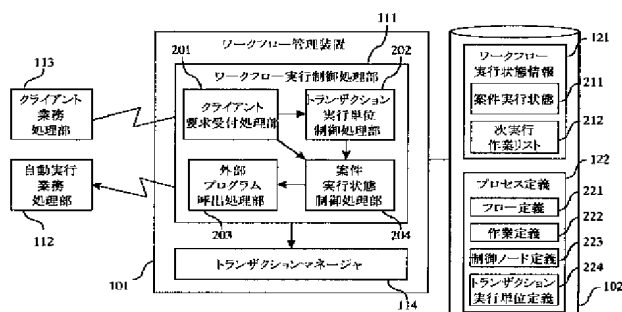
(54)【発明の名称】 ワークフロー管理方法及びその実施装置並びにその処理プログラムを記録した記録媒体

(57)【要約】

【課題】 ワークフローによって処理される複数の作業の信頼性を高めることが可能な技術を提供する。

【解決手段】 一連の処理順序に従って実行される複数の作業の実行管理を行うワークフロー管理方法において、処理要求を受付けて一連の作業の処理手続を開始するステップと、複数作業に渡る処理を同一トランザクションで行う際に、同一トランザクションで実行すべき作業単位を決定し、トランザクションの開始とコミットの制御を行うステップと、作業の進行状態の制御及び管理を行うステップとを有するものである。

図 4



【特許請求の範囲】

【請求項1】 一連の処理順序に従って実行される複数の作業の実行管理を行うワークフロー管理方法において、

処理要求を受付けて一連の作業の処理手続を開始するステップと、複数作業に渡る処理を同一トランザクションで行う際に、同一トランザクションで実行すべき作業単位を決定し、トランザクションの開始とコミットの制御を行うステップと、作業の進行状態の制御及び管理を行うステップとを有することを特徴とするワークフロー管理方法。

【請求項2】 同一のトランザクションとして実行する複数作業を決定する定義情報をプロセス定義内に含むことを特徴とする請求項1に記載されたワークフロー管理方法。

【請求項3】 複数作業を行うトランザクション内でワークフロー実行状態情報に対する更新処理を行うことを特徴とする請求項1または請求項2のいずれかに記載されたワークフロー管理方法。

【請求項4】 一連の処理順序に従って実行される複数の作業の実行管理を行うワークフロー管理装置において、

処理要求を受付けて一連の作業の処理手続を開始する要求受付処理部と、複数作業に渡る処理を同一トランザクションで行う際に、同一トランザクションで実行すべき作業単位を決定し、トランザクションの開始とコミットの制御を行うトランザクション実行単位制御処理部と、作業の進行状態の制御及び管理を行う案件実行状態制御処理部とを備えることを特徴とするワークフロー管理装置。

【請求項5】 一連の処理順序に従って実行される複数の作業の実行管理を行うワークフロー管理装置としてコンピュータを機能させる為のプログラムを記録したコンピュータ読み取り可能な記録媒体において、処理要求を受付けて一連の作業の処理手続を開始する要求受付処理部と、複数作業に渡る処理を同一トランザクションで行う際に、同一トランザクションで実行すべき作業単位を決定し、トランザクションの開始とコミットの制御を行うトランザクション実行単位制御処理部と、作業の進行状態の制御及び管理を行う案件実行状態制御処理部としてコンピュータを機能させる為のプログラムを記録したことを特徴とする記録媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は複数の作業からなる業務の流れを管理するワークフロー管理装置に関し、特にワークフロー管理システムにおいて複数の業務プログラム（アプリケーションプログラム）の実行を制御する際に、複数作業における複数業務処理の間でトランザクション処理を行うワークフロー管理装置に適用して有効

な技術に関するものである。

【0002】

【従来の技術】ワークフロー管理システムは、複数の作業からなる業務の流れをプロセス定義として事前定義し、この定義に基づいて作業実行と業務の進行状態の管理を支援する技術である。各作業における処理は、人が行う場合や、業務プログラムにより自動的に実行される場合があるが、ワークフロー管理システムはこれらの作業の人への割当てや自動実行業務処理部の実行を行い、且つ業務全体としての進行管理を行うものである。

【0003】ワークフロー管理システムは、主にホワイトカラーの生産性向上を目的とし、オフィス業務の手続をコンピュータを用いて自動化すると言った観点から発展してきた（速水治夫、「ワークフロー・ソフトの基本原則－動作の仕組みと導入効果」、日経コンピュータ、1997年9月1日号、pp. 204-217.）。

【0004】更に近年、ワークフロー管理システムの技術を、業務プログラムの実行制御や業務システム間の連携に用いたり、より信頼性を要求されるミッションクリティカルな業務へ適用したり、と言った事が行われている。こうした点から、ワークフロー管理システムにおける信頼性の向上や、業務プログラムとワークフロー実行制御処理の間での柔軟なトランザクション制御を可能とする技術が重要となってきている。

【0005】従来、複数の計算機が通信ネットワークで接続された分散システムにおいて、異なる計算機にわたる処理をトランザクション特性（ACID特性と呼ばれる）を保ちながら行う方法として、2相コミットを用いた分散トランザクション処理技術が広く一般的に用いられている（「トランザクション処理システム入門」、日経BP社）。

【0006】特開平8-153068号公報「ワークフローシステム」では、こうした分散トランザクション処理技術をワークフロー管理システムに取り入れ、業務プログラム処理による業務データ更新とワークフロー実行制御処理の管理するワークフロー実行状態情報とを同一トランザクションで一貫性を保って更新する方法を示している。

【0007】しかしこの従来技術は、業務処理における業務データの更新とワークフロー実行制御処理の管理するワークフロー実行状態情報の更新間でのトランザクション保証が目的であり、複数の異なった作業における業務処理間でのトランザクション保証を目的としたものではない。

【0008】

【発明が解決しようとする課題】ワークフロー管理システムの提供する利点の一つとして、業務手順の変更に対するシステムの柔軟性が上げられる。これは、プロセス定義で与えられる業務全体のマクロな処理の流れと、各作業において実行される業務処理とを分離する事で、各

作業で実行される業務プログラムの部品としての機能の独立性を高め、その結果として業務プログラムの再利用性が高められる事から得られる。

【0009】ワークフロー管理システムを用いた業務プログラムの実行制御や業務システム連携において、個々の業務プログラム内にトランザクション制御の為にロジック（分散トランザクションの開始やコミット）を記述した場合、個々の業務プログラムの処理は夫々個別のトランザクションとしてしか実行できず、これらの業務プログラムを組合せた柔軟なトランザクション制御が出来

10

ないと言う問題がある。この問題は、個々の業務プログラムの処理粒度が小さく、これらのプログラムの組合せによってシステム構築を行う際には、より重大となる。

【0010】例えば、オンラインで商品を注文し、銀行の口座から預金の一部を出金した後、出金した金額を他の銀行に送金し、送金された金額を送金先の特定の口座に入金するといった一連の複数の作業をワークフロー管理システムで実行すれば、作業の順番が変更された場合にも柔軟に対応することができるが、途中の作業でエラーが発生した場合には、それ以前の処理を取り消さなければ一貫性を保つことができないという問題がある。

20

【0011】本発明の目的は上記問題を解決し、ワークフローによって処理される複数の作業の信頼性を高めることが可能な技術を提供することにある。本発明の他の目的は複数作業を同一のトランザクションで実行する際に、個々の業務プログラムを変更すること無く業務プログラムの組合せを柔軟に変更することが可能な技術を提供することにある。本発明の他の目的は業務プログラムの部品としての独立性を高め再利用し易くすることが可能な技術を提供することにある。本発明の他の目的は複数作業における複数業務プログラム処理とそのワークフローの実行管理の一貫性を高めることが可能な技術を提供することにある。

30

【0012】

【課題を解決するための手段】本発明は、一連の処理順序に従って実行される複数の作業の実行管理を行うワークフロー管理装置において、ワークフローによって処理される複数の作業を同一トランザクションとして実行するものである。

【0013】本発明のクライアント要求受付処理部が通信ネットワークを介してクライアント業務処理部からの処理要求を受付けると、トランザクション実行単位制御処理部は、プロセス定義に基づいて同一トランザクションで実行すべき作業単位を決定し、複数作業に渡る業務処理を同一トランザクションで行う際にそのトランザクションを開始させる。

【0014】外部プログラム呼出処理部は、前記開始されたトランザクションの作業に対して割当てられた自動実行業務処理部の呼出しを通信ネットワーク経由で行い、その作業を業務サーバや業務クライアントで実行さ

50

せる。

【0015】案件実行状態制御処理部は、プロセス定義に基づいて作業の進行状態の制御及び管理を行い、同一のトランザクションで実行すべき複数の作業の実行が完了すると、トランザクション実行単位制御処理部により、当該トランザクションのコミット処理を行う。

【0016】以上の様に本発明のワークフロー管理装置によれば、ワークフローによって処理される複数の作用を同一トランザクションとして実行するので、ワークフローによって処理される複数の作業の信頼性を高めることが可能である。

【0017】

【発明の実施の形態】以下に一連の処理順序に従って実行される複数の作業の実行管理を行う一実施形態のワークフロー管理装置について説明する。本実施形態では、説明が煩雑になる事を避ける為、一般的なワークフロー管理システムに比較して単純化され、また、本実施形態の説明に不必要な機能は省略されている。

【0018】図1は本実施形態のワークフロー管理システムの構成例を示す図である。本実施形態では、ワークフロー管理装置101、業務サーバ104、業務クライアント105の各計算機及びデータベース102は通信ネットワーク103により接続されている。各計算機は、分散トランザクションの管理及び制御を行うトランザクションマネージャ114を備えており、分散トランザクションのトランザクション特性の保証は、これらのトランザクションマネージャ114及びデータベース102間の連携により、適切に行われる。

【0019】ワークフロー管理装置101のワークフロー実行制御処理部111は、データベース102内に格納されたプロセス定義122を参照し、その定義内容に基づいて作業の流れの進行状態の管理や作業に対応付けられた自動実行業務処理部112の呼出し、クライアント業務処理部113からの処理要求の受付け、及びトランザクションの実行単位制御を行う。作業の流れの進行状態は、ワークフロー実行状態情報121としてデータベース102内に格納される。

【0020】自動実行業務処理部112及びクライアント業務処理部113は、各作業における業務処理を実行する。自動実行業務処理部112は、ワークフロー実行制御処理部111により通信ネットワーク103を介して呼出され、同期的に業務処理を実行する。

【0021】クライアント業務処理部113は、業務処理の実行を開始可能な作業において、ワークフロー実行制御処理部111の処理とは非同期に業務処理を行った後、通信ネットワーク103を介してワークフロー実行制御処理部111へ作業の完了通知を行う。またクライアント業務処理部113は、例えば作業に割当てられその実行に対して責務を負う作業者によって起動され、操作される事で動作する。クライアント業務処理部113

は、その他にも一連の作業の流れ全体を開始する案件投入と呼ぶ処理の実行要求や、実行開始可能な作業リストの取得要求をワークフロー実行制御処理部111に対して行う。

【0022】これらの自動実行業務処理部112及びクライアント業務処理部113の処理において参照及び更新を行う業務データ123は、データベース102に格納される。

【0023】業務データ123及びワークフロー実行状態情報121に対する参照及び更新処理は、トランザクションマネージャ114及びデータベース102の連携によりトランザクション特性を保証しつつ実行される。この為に、分散トランザクションを一意に特定する目的でトランザクションマネージャ114が発行し管理するトランザクション記述子を、データベース102へのアクセスを行うこれらのプログラム間で持ち回り、データベース102に格納されたデータに対して参照及び更新処理を行う際には、データベース102に対して受け渡す等の方法が一般的に取られる。

【0024】即ち、ワークフロー実行制御処理部111、自動実行業務処理部112、クライアント業務処理部113の各データベースアクセスプログラムが、データベース102に対して参照更新処理を行う際に、トランザクション記述子がデータベース102へ渡される。但し、本実施形態の説明が煩雑になる事を防ぐ為、自明な場合にはトランザクション記述子がデータベース102へ受け渡されている旨を明記する事はしないものとする。また、トランザクション記述子は分散トランザクション管理そのものを実現する為のものであって、本実施形態の適用範囲がこれにより制限されるものではない。

【0025】プロセス定義ツール115は、ワークフロー実行制御処理部111が作業の流れを制御するのに参照するプロセス定義122を、ワークフロー管理システムの運用者であるユーザが編集して作成する際にその処理を支援するものである。

【0026】図2は本実施形態の各計算機の概略構成を示す図である。図2では本実施形態における各計算機の一般的な構成を表しており、各計算機は、表示装置3001、CPU3002、通信装置3003、キーボード3004、マウス3005、メモリ3006、補助記憶装置3007を備えている。ワークフロー実行制御処理部111、自動実行業務処理部112、クライアント業務処理部113、トランザクションマネージャ114、プロセス定義ツール115は、例えば、ソフトウェアプログラムとして実装され、補助記憶装置3007に格納されている。これらのソフトウェアプログラムは、メモリ3006に読み込まれ、CPU3002により実行される。また、表示装置3001やキーボード3004、マウス3005の入力装置を用いて、ワークフロー管理システムの利用者であるユーザとの対話を行ったり、通

信装置3003を用いて他の計算機上で動作するソフトウェアプログラムやデータベース102と通信を行う事が出来る。

【0027】図3は本実施形態の動作原理の概略を示す図である。ここで、本実施形態の基本的な動作原理の概略について上記の構成及び図3を用いて説明する。本実施形態のワークフロー管理システムのユーザは、プロセス定義ツール115を用いて、一連の作業の処理順序をプロセス定義122として定義する。この時、1つのトランザクションとして処理する複数作業のまとまりを決定する定義情報を、プロセス定義内に併せて定義する事で上記定義情報をワークフロー実行制御処理部111に与える事が出来る。

【0028】ワークフロー実行制御処理部111は、プロセス定義122を基に、複数作業に対する処理を行う自動実行業務処理部112を順に実行する。この際、1つのトランザクションとして処理する複数作業の実行前にトランザクションを開始し（トランザクション開始要求4003）、自動実行業務処理部112を順に呼出す事で上記複数作業に対する処理を実行し（業務プログラム呼出し4001）、最後に上記複数作業に対する処理を纏めて1つのトランザクションとしてコミット（トランザクションコミット処理要求4004）する。

【0029】図3の例では、作業Aと作業Bの組及び作業Cと作業Dの組のそれぞれを、1つのトランザクションとして実行している。ここで、ワークフロー実行状態情報121の更新処理4002を各作業に対する自動実行業務処理部112の処理と同じトランザクションで行う事で、業務データ123とワークフロー実行状態情報121の更新処理に対するトランザクション特性を保証する事も出来る。

【0030】また、ワークフロー実行状態情報121と業務データ123との更新処理に対するトランザクション保証が不要な場合、ワークフロー実行状態情報121をメモリ3006内に格納し、トランザクションとは無関係に管理し更新する実装も可能である。複数の業務データ123間及びワークフロー実行状態情報121との間のトランザクション保証は、データベース102及びトランザクションマネージャ114の間で適切に行われる。

【0031】以下、本実施形態のより詳細な説明を行う。図4は本実施形態のワークフロー管理装置101の概略構成を示す図である。図4ではワークフロー実行制御処理部111と、ワークフロー実行制御処理部111が参照及び更新を行うワークフロー実行状態情報121及びプロセス定義122の構成を表している。

【0032】ワークフロー実行制御処理部111は、クライアント要求受付処理部201、トランザクション実行単位制御処理部202、外部プログラム呼出処理部203、そして案件実行状態制御処理部204を有してい

る。

【0033】クライアント要求受付処理部201は、通信ネットワーク103を介してクライアント業務処理部113からの処理要求を受付ける機能を提供する。トランザクション実行単位制御処理部202は、複数作業に渡る業務処理を同一トランザクションで行う際に、プロセス定義122に基づいて同一トランザクションで実行すべき作業単位を決定し、トランザクションの開始とコミットの制御を行う。

【0034】案件実行状態制御処理部204は、プロセス定義122に基づいて、作業の進行状態の制御及び管理を行う機能を提供する。外部プログラム呼出処理部203は、作業に対して割当てられた自動実行業務処理部112の呼出しを、通信ネットワーク103を介して行う機能を提供する。

【0035】ワークフロー管理装置101を、クライアント要求受付処理部201、トランザクション実行単位制御処理部202、外部プログラム呼出処理部203、案件実行状態制御処理部204として機能させる為のプログラムはCD-ROM等の記録媒体に記録され、磁気ディスク装置に格納された後、メモリにロードされて実行されるものとする。なお前記プログラムを記録する記録媒体はCD-ROM以外の他の記録媒体でも良い。

【0036】クライアント要求受付処理部201は、新規に一連の作業の処理手続を開始する(案件投入)「Start」、特定の作業の完了通知を受付ける「Complete」、作業に対する処理を実行開始可能な案件のリストを取得する「GetWorkList」の3種類の手続を有し、それぞれに対するクライアント業務処理部113からの処理要求を受付ける。

【0037】プロセス定義122は、フロー定義221、作業定義222、制御ノード定義223、トランザクション実行単位定義224を有している。作業定義222は、実行すべき一連の作業のリストと、その定義属性を含んでいる。制御ノード定義223は、作業の流れの開始点や終了点及び分岐点や合流点を示す制御ノードと呼ぶ定義要素のリストと、その定義属性を含んでいる。フロー定義221は、各作業及び制御ノード間の関連付けに関する情報を含み、作業の流れの定義情報を与える。トランザクション実行単位定義224は、複数の作業の処理に渡るトランザクション実行において、同一トランザクションで処理すべき作業の実行単位を決定する為に用いる定義情報を含んでいる。

【0038】ワークフロー実行状態情報121は、案件実行状態211、次実行作業リスト212を有している。案件実行状態211は、各作業の処理の実行状態、及び各制御ノードでのフローの制御情報を含んでいる。次実行作業リスト212は、ワークフロー実行制御処理部111の処理のある時点において、次に自動実行すべき作業のリストを含んでいる。これらの実行状態情報

は、案件と呼ばれる一連の作業の流れの組毎に保持される。

【0039】以下、本実施形態におけるプロセス定義122の各構成要素の詳細な説明と、具体的なプロセス定義例を示す。図5は本実施形態の作業定義222の含む定義項目と具体的な定義情報の例を示す図である。作業定義222は、作業の名称を示すノード名401、作業の実行方法の種別を示す種別402、作業の処理において自動実行業務処理部112を実行するのに用いるプログラム情報403を含むテーブルである。作業の種別402は、自動実行業務処理部112を同期的に呼出して作業に対する処理を行った後にその作業を完了状態にする「同期自動実行」、自動実行業務処理部112を同期的に呼出して作業に対する処理を行った後にその作業に対する完了通知をクライアント業務処理部113から非同期に受取る「非同期自動実行」、作業に割当てられた作業者がクライアント業務処理部113を実行して業務処理を行った後にその作業に対する完了通知をワークフロー実行制御処理部111に対して行う「作業実行」の3種類の何れかである(「同期自動実行」及び「非同期自動実行」の双方の作業を、自動実行作業と呼ぶ)。プログラム情報403は、通信ネットワーク103を介してワークフロー実行制御処理部111が自動実行業務処理部112を呼出すのに必要な情報を含んでいる。

【0040】図6は本実施形態の制御ノード定義223の含む定義項目と具体的な定義情報の例を示す図である。制御ノード定義223は、制御ノードの名称を示すノード名501、及び制御ノードの種別を含む種別502を含むテーブルである。この例では、制御ノードの種別502として、作業の処理の流れの開始点を示す「Start」、終了点を示す「End」、分岐点を示す「Split」、合流点を示す「Join」を用いている。作業定義222におけるノード名401及び制御ノード定義223におけるノード名501は、プロセス定義122において一意となる名前が付けられる。

【0041】図7は本実施形態のフロー定義221の含む定義項目と具体的な定義情報の例を示す図である。フロー定義221は、作業及び制御ノード(単にノードと呼ぶ)間の処理の流れの順序関係を示しており、遷移元301及び遷移先302を有している。

【0042】図8は本実施形態のトランザクション実行単位定義224の含む定義項目と具体的な定義情報の例を示す図である。トランザクション実行単位定義224は、同一のトランザクション実行単位に含まれる作業の集まりを示す実行単位作業リスト602、及びトランザクションの実行単位をプロセス定義内で一意に特定する実行単位ID601を有している。トランザクション実行単位制御処理部202は、同一のトランザクション実行単位に含まれる複数の作業を、一つのトランザクシ

ンとして処理する様にトランザクションの制御を行う。

【0043】図7～図8におけるプロセス定義の例は、作業「A」～作業「E」の5つの作業を含み、作業「A」～「C」を逐次的に処理した後、作業「D」～「E」を並行して処理する様に実行される。また、作業「A」～「C」の処理、及び作業「D」～「E」の処理を夫々同一トランザクションで実行する様に制御する。

【0044】プロセス定義は、プロセス定義ツール115を用いて編集作成する事が出来る。図9を用いて、プロセス定義ツール115を用いた定義方法の例を示す。図9は本実施形態のプロセス定義ツール115の表示画面を示す図である。図9では、プロセス定義ツール115が図2における表示装置3001の画面に表示するイメージを表している。また、プロセス定義ツール115は、キーボード3004により文字や記号の入力を受け付けたり、マウス3005を用いて画面上の一点を示すマウスポインタ2004を操作する事が出来る。ユーザはマウスポインタ2004を用いて直接表示イメージを操作する事で、プロセス定義の編集作成を行う事が出来る。こうした方法自体は、GUI（グラフィカルユーザインタフェース）と呼ばれる一般的に知られた技法である。

【0045】プロセス定義ツール115の表示画面は、大きく分けるとレイアウトボックス2001及びツールボックス2002に分けられる。ツールボックス2002には、プロセス定義の各構成要素に対応した定義部品2011～2017が表示されている。ユーザは、マウスポインタ2004を用いてツールボックス2002から必要な定義部品を選択し、（ドラッグアンドドロップ等の操作により）レイアウトボックス2001内にコピーして配置する事ができる。

【0046】先ず、ユーザは作業の定義部品2012や制御ノードの定義部品2013～2016をレイアウトボックス2001内にコピーして配置する。作業2022はコピーされた作業の定義部品であり、制御ノード2023～2026はコピーされた制御ノードの定義部品である。

【0047】更に、ユーザは、アローの定義部品2011をレイアウトボックス2001内にコピーし、コピーされた作業2022や制御ノード2023～2026を接続する。アロー2021はコピーされたアローの定義部品である。

【0048】コピーされた各定義部品毎のより詳細な定義属性は、定義属性設定ダイアログボックス2003により設定する事が出来る。定義属性設定ダイアログボックス2003は、例えばマウスポインタ2004を用いて定義属性を設定したい定義部品のコピーを（ダブルクリック等の操作により）選択する事で表示される。

【0049】ユーザは、再びマウスポインタ2004を用いて、定義属性設定ダイアログボックス2003内の

設定したい属性を選択し、キーボード3004により文字や記号を入力する。

【0050】次に、同一トランザクションで処理すべき作業の単位を、トランザクション実行単位の定義部品2017を用いて定義する。ユーザは、トランザクション実行単位の定義部品2017をレイアウトボックス2001内にコピーする。更にコピーされたトランザクション実行単位2027の位置や大きさをマウスポインタ2004を用いて変更し、同一のトランザクション内で処理する作業の集合を囲む様に配置する。図9では、作業「A」～「C」が同一トランザクションで実行される様に、トランザクション実行単位2027により囲まれている。また、作業「D」～「E」についても同様である。こうして、プロセス定義ツール115を用いてユーザが入力した定義情報は、データベース102内にプロセス定義122として格納される。

【0051】図10は本実施形態のプロセス定義ツール115の保持する定義情報をプロセス定義122として格納する処理の処理手順を示すフローチャートである。先ず、ステップ1801において、プロセス定義122内の全てのテーブルを初期化する（実施の仕方によっては、複数のプロセス定義の管理も可能だが、本実施形態の本質とは無関係に煩雑になる事を避ける為、単一のプロセス定義のみを扱う様にする）。

【0052】ステップ1802において、レイアウトボックス2001内に配置された全ての作業2022に対するレコードを、作業定義222へ挿入する。この時、作業定義222の各定義要素（ノード名401、種別402、プログラム情報403）には、定義属性設定ダイアログボックス2003を用いて入力された作業の定義属性を設定する。

【0053】次にステップ1803において、レイアウトボックス2001内に配置された全ての制御ノード2023～2026に対するレコードを、制御ノード定義223へ挿入する。この時、制御ノード定義223の定義属性（ノード名501）には、定義属性設定ダイアログボックス2003を用いて入力された制御ノードの定義属性を設定する。また、種別502は、制御ノード2023～2026の種別に対応した値を設定する。

【0054】次にステップ1804において、レイアウトボックス2001内に配置された全てのアロー2021に対するレコードを、フロー定義221へ挿入する。この時、アローの根元側に位置するノードのノード名を遷移元301に、アローの指示す側に位置するノードのノード名を遷移先302に設定する。

【0055】最後に、ステップ1805において、レイアウトボックス2001内に配置された全てのトランザクション実行単位2027に対するレコードをトランザクション実行単位定義224へ挿入する。この時、実行単位ID601は、プロセス定義内で一意となる識別子

を自動採番して設定する。また、実行単位作業リスト602には、トランザクション実行単位2027で囲まれた全ての作業のノード名のリストを設定する。以上で、プロセス定義122の生成が完了する。図7～図8に示したプロセス定義の具体例は、図9のレイアウトボックス2001内に示したプロセス定義例と対応している。

【0056】こうしたプロセス定義ツールは、ワークフロー管理システムにおいては一般的であり、本実施形態の特徴であるトランザクション実行単位定義224もGUIを用いて作成することができる。

【0057】以下に、図11及び図12を用いて、ワークフロー実行状態情報121の詳細な構成について説明する。図11は本実施形態の案件実行状態211の項目を示す図である。案件実行状態211は、各案件毎に全ての作業及び制御ノードの実行状態を含んでいる。案件キー701は、案件を一意に特定する識別子であり、案件キーを用いて案件実行状態211のテーブルを検索する事で、その案件における全てのノード（作業及び制御ノード）の状態を取得する事が出来る。ノード名702には、作業のノード名401若しくは制御ノードのノード名501の値が格納されており、何れのノードのレコードかを特定するのに用いる。状態703は、ノードの現在の実行状態を表し、ノードの種別毎に異なる値が格納される。ノードが作業の場合、作業の実行状態には、「Initial」、「Ready」、「Completed」の3つの状態があり、「Initial」は作業に対する処理が未着手で且つ実行開始不可能な状態、「Ready」は作業に対する処理が実行中か実行開始可能な状態、「Completed」は作業に対する処理の実行が完了した状態を表している。ノードが合流点制御ノード（「Join」）の場合には、処理の流れの合流点への到達数が格納される。

【0058】図12は本実施形態の次実行作業リスト212の項目を示す図である。次実行作業リスト212は、次に実行すべき自動実行作業の作業リスト802を案件毎に保持している。作業リスト802は案件キー801を用いて、案件毎に一意に特定出来る。

【0059】次に、ワークフロー実行制御処理部111の各構成要素の動作を、適宜フローチャートを用いて説明する。先ず、図13、図14を用い、クライアント要求受付処理部201の処理手順を説明する。前述した様に、クライアント要求受付処理部201は、新規に一連の作業の処理手続を開始する（案件投入）「Start」、特定の作業の完了通知を受付ける「Complete」、作業に対する処理を実行開始可能な案件のリストを取得する「GetWorkList」の3種類の手続を有し、それぞれに対するクライアント業務処理部113からの処理要求を受付ける。

【0060】図13は本実施形態の「Start」手続の処理手順を示すフローチャートである。「Star

」手続は、呼出しパラメータとしてクライアント業務処理部113から案件キー及びトランザクション記述子を受取る。トランザクション記述子を受取る事で、クライアント業務処理部113の行う業務処理と、ワークフロー実行制御処理部111の行う案件投入処理とを、同一トランザクションとして実行する事を可能としている。この必要が無い場合、クライアント業務処理部113は、トランザクション記述子として無効な値「NULL」を受け渡す事で、案件投入処理の為にトランザクションをワークフロー実行制御処理部111側で開始する様に指示する事が出来る。

【0061】また、案件キーは、システム内で一意な識別子であり、システム内で案件を特定したり、業務データとの関連付けを行うのに用いる。本実施形態では、クライアント業務処理部113において案件キーの採番を行う。

【0062】「Start」手続の処理手順は以下の様になる。先ず、最初のステップ1001において、呼出しパラメータとして案件キーとトランザクション記述子を受取る。次に、ステップ1002において、トランザクション記述子の値が「NULL」か否かを検査する。「NULL」の場合は、トランザクションマネージャ114に対して新規にトランザクションの開始を要求し、開始したトランザクションのトランザクション記述子を受取る（ステップ1003）。

【0063】ステップ1004において、案件キー及びトランザクション記述子をパラメータとして案件実行状態制御処理部204の手続「StartProcess」を呼出す。「StartProcess」手続は、新たに開始する案件に関する初期処理を行い、実行状態や制御情報をワークフロー実行状態情報121へ格納する。この手続については、案件実行状態制御処理部204の処理手順の説明時に詳細に説明する。

【0064】次に、ステップ1005において、トランザクションマネージャ114に対して、現在の処理トランザクションのコミットを要求する。この段階で、クライアント業務処理部113の行った業務処理と、案件投入処理とが、同一トランザクション処理としてコミットされる。

【0065】更に、ステップ1006において、クライアント業務処理部113へ「Start」要求に対する処理の完了を通知する（クライアント要求受付処理部201の各手続の通信ネットワーク103を介した実行が、リモートプロシージャコールの様な同期手続呼出しのモデルで実装される場合、この時点でクライアント業務処理部113へ制御が戻る）。この時点で、クライアント業務処理部113は、案件投入処理が確実に行われた事を確認する事が出来る。

【0066】最後に、ステップ1007において、案件キーをパラメータとしてトランザクション実行単位制御

10

20

30

40

50

処理部202の手続「ExecuteSucceedingTransaction」手続を呼出す。「ExecuteSucceedingTransaction」手続は、案件投入処理や作業の完了処理の結果発生した、別トランザクションに分けて実行すべき自動実行作業の処理を、新規にトランザクションを開始して行う。この手続については、トランザクション実行単位制御処理部202の処理手順の説明時に詳細に説明する。以上で、クライアント要求受付処理部201の「Start」手続の処理を完了する。

【0067】図14は本実施形態の「Complete」手続の処理手順を示すフローチャートである。「Complete」手続は、呼出しパラメータとしてクライアント業務処理部113から、案件キー、作業のノード名、及びトランザクション記述子を受取る。トランザクション記述子を受取る事で、作業に対応してクライアント業務処理部の行った業務処理と、作業の完了処理、及び同一トランザクション実行単位内に含まれる他の作業の処理とを、同一のトランザクションとして実行する事を可能としている。この必要が無い場合、クライアント業務処理部113は、トランザクション記述子として無効な値「NULL」を受け渡す事で、作業完了処理の為のトランザクションをワークフロー実行制御処理部111側で開始する様に指示する事が出来る。

【0068】「Complete」手続の処理手順は以下の様に成る。まず、最初のステップ1101において、呼出しパラメータとして、案件キー、作業のノード名、及びトランザクション記述子を受取る。

【0069】次に、ステップ1102において、トランザクション記述子の値が「NULL」か否かを検査する。「NULL」の場合は、トランザクションマネージャ114に対して新規にトランザクションの開始を要求し、開始したトランザクションのトランザクション記述子を受取る（ステップ1103）。

【0070】ステップ1104において、案件キー、作業のノード名、及びトランザクション記述子をパラメータとして案件実行状態制御処理部204の手続「CompleteWork」を呼出す。「CompleteWork」手続は、指定された案件キー及びノード名の作業の完了処理を行う。この手続については、案件実行状態制御処理部204の処理手順の説明時に詳細に説明する。

【0071】次に、ステップ1105において、案件キー、作業のノード名、及びトランザクション記述子をパラメータとして、トランザクション実行単位制御処理部202の手続「CompleteCurrentTransaction」を呼出す。「CompleteCurrentTransaction」手続は、パラメータとして受け渡したノード名の作業と同一のトランザクションで行うべき一連の自動実行作業に対する処理を実

行する。この手続については、トランザクション実行単位制御処理部202の処理手順の説明時に詳細に説明する。

【0072】次に、ステップ1106において、トランザクションマネージャ114に対して、現在の処理トランザクションのコミットを要求する。この段階で、クライアント業務処理部113の行った業務処理と、作業の完了処理、及び同一トランザクション実行単位内に含まれる他の作業の処理とが、同一のトランザクション処理としてコミットされる。

【0073】更に、ステップ1107において、クライアント業務処理部113へ「Complete」要求に対する処理の完了を通知する。この時点で、クライアント業務処理部113は、作業完了処理が確実に行われた事を確認する事が出来る。最後に、ステップ1108において、案件キーをパラメータとしてトランザクション実行単位制御処理部202の手続「ExecuteSucceedingTransaction」手続を呼出す。以上で、クライアント要求受付処理部201の「Complete」手続の処理を完了する。

【0074】クライアント要求受付処理部201は、更に「GetWorkList」手続を有している。「GetWorkList」手続は、作業のノード名を呼出しパラメータとして受取り、案件実行状態211においてノード名702の値がパラメータで指定されたノード名に等しく且つ状態703が「Ready」のレコードを検索し、それらのレコードの案件キー701の値のリストを結果として返す。

【0075】次に、図15、図17、図18を用いて、案件実行状態制御処理部204の処理手順を説明する。案件実行状態制御処理部204は、案件投入処理時に案件実行状態の初期処理及び開始処理を行う「StartProcess」手続、作業完了時に作業の状態703の更新処理及びフロー定義221に従って引続いて実行すべき作業の決定処理を行う「CompleteWork」手続、作業に対応付けられた自動実行業務処理部112の呼出し処理を行う「ExecuteWork」手続を有している。図15に「StartProcess」手続の処理手順を、図17に「CompleteWork」手続の処理手順を、図18に「ExecuteWork」手続の処理手順を示す。

【0076】図15は本実施形態の「StartProcess」手続の処理手順を示すフローチャートである。「StartProcess」手続は、呼出しパラメータとして案件キー及びトランザクション記述子を受取る。処理手順は以下の様になる。

【0077】まず、最初のステップ1201において、呼出しパラメータとして案件キーとトランザクション記述子を受取る。次に、ステップ1202において、作業定義222及び制御ノード定義223を基に、新規案件

10

20

30

40

50

に対する案件実行状態211を初期化する。即ち、作業定義222に格納された全てのレコード及び制御ノード定義223に格納された全てのレコードに1対1に対応するレコードを案件実行状態211へ挿入する。

【0078】この時、案件キー701にはステップ1201でパラメータとして受取った案件キーを、ノード名702には作業定義222の各レコードのノード名401及び制御ノード定義223の各レコードのノード名501を設定する。更に、全ての作業に対するレコードにおいて状態703を「Initial」で初期化し、全ての「Join」制御ノードにおいて状態703を「0」で初期化する。

【0079】次に、ステップ1203において、フロー定義221を用いて「Start」制御ノードから遷移する最初に実行すべき作業を決定し、そのノード名のリストを作成する。この処理は、フロー定義221における遷移元301から遷移先302を辿る事で行われるが、詳細については後述する。

【0080】ステップ1204において、これらの作業に対する状態703を「Ready」へ変更する。更に、ステップ1205において、作業定義222を参照して、ステップ1203で作成した作業リストの内で作業の種別402が自動実行作業（「同期自動実行」及び「非同期自動実行」）である作業を決定し、そのノード名リストを作成する。

【0081】ステップ1206では、ステップ1205で作成した作業のノード名リストを作業リスト802に、ステップ1201で受取った案件キーを案件キー801に設定したレコードを次実行作業リスト212へ挿入する。

【0082】図16は本実施形態の「TransitionSolver」手続の処理手順を示すフローチャートである。次にステップ1203の処理の詳細を示す。図16のフローチャートは、ステップ1203を実行するのに用いる「TransitionSolver」手続の処理手順を表しており、「TransitionSolver」手続は、案件キー、ノード名、及びトランザクション記述子をパラメータとして受取り、ノード名のリストを結果として返す。「TransitionSolver」手続は、手続内でその手続自身を呼出す為、再帰的に実行される。

【0083】ステップ1203においては、「TransitionSolver」手続を呼出す前に、次の準備を行う。即ち、制御ノード定義223を検索し、種別502が「Start」の制御ノードのノード名501を特定する。このノード名を、「TransitionSolver」手続の2番目のパラメータとして渡し、手続を呼出す。呼出しの結果、「TransitionSolver」から受取った結果リストが、ステップ1203の結果として得られる作業のノード名のリストと

なる。

【0084】それでは、図16を用いて、「TransitionSolver」手続の詳細な処理手順を説明する。まず、ステップ1901において、案件キー、ノード名、及びトランザクション記述子をパラメータとして受取る。

【0085】ステップ1902において、フロー定義221を検索し、遷移元301の値がステップ1901で受取ったノード名と等しいレコードを特定し、それらのレコードの遷移先302に格納されたノード名のリスト（遷移先ノードリストと呼ぶ）を作成する。ステップ1903において、遷移先ノードリストから任意の1つを選択し、リストから取り除く。

【0086】次に、ステップ1904において、作業定義222及び制御ノード定義223を用い、ステップ1903で選択したノード名のノードの種別を特定する。選択したノードの種別が作業の場合（ステップ1905）、ノード名を結果リストに追加する（ステップ1906）。結果リストは、この手続の呼出し毎に生成され、呼出し時に空のリストとして初期化され、この手続の完了時に結果として呼出し側へ返される。

【0087】選択したノードの種別が「Join」制御ノードの場合（ステップ1907）、案件キー及びノード名から案件実行状態211を検索し、「Join」制御ノードの状態703を取得し、取得した値に「1」を加算した数を再度状態703へ書き戻す。

【0088】更に、遷移先302の値が選択したノード名に等しいフロー定義221のレコードを検索し（「Join」制御ノードへの全ての入力遷移を調べる）、そのレコード数と状態703の値を比較する（ステップ1909）。値が等しい場合、その「Join」制御ノードに対する、全ての入力遷移が行われた事を示し、更に「Join」制御ノードの次の遷移先を調べる為にステップ1911を実行する。選択したノードの種別が「Split」の場合（ステップ1910）も、同様に更に次の遷移先を調べる為にステップ1911を実行する。

【0089】ステップ1911では、選択したノード名を2番目のパラメータとして、「TransitionSolver」手続を再帰呼出しする。再帰呼出しの結果得られた結果リストは、現在の「TransitionSolver」手続の呼出しにおける結果リストに付け加える（ステップ1912）。

【0090】ステップ1903から、ステップ1913の間の処理は、ステップ1902で作成した遷移先ノードリストが空になるまで繰り返され、結果として得られた作業のノード名のリスト（結果リスト）がこの手続の呼出し元へ返される（ステップ1914）。「TransitionSolver」の具体的な動作については、後述する具体的なプロセス定義の説明において明らかにする。

【0091】図17は本実施形態の「Complete Work」手続の処理手順を示すフローチャートである。「CompleteWork」手続は、呼出しパラメータとして、案件キー、作業のノード名、及びトランザクション記述子を受取る。処理手順は以下の様になる。

【0092】まず、最初のステップ1301において、呼出しパラメータとして案件キー、作業のノード名、及びトランザクション記述子を受取る。次に、ステップ1302において、パラメータとして受取った案件キー及び作業のノード名から案件実行状態211を検索し、特定したレコードの状態703の値を「Completed」へ更新する。

【0093】次に、ステップ1303において、フロー定義221を用いて、完了した作業から遷移する次に実行すべき作業を決定し、そのノード名のリストを作成する。このステップ1303の処理は、「TransitionSolver」手続を用いて、ステップ1203と同様に行われる。

【0094】次に、ステップ1304において、これらの作業に対する状態703を「Ready」へ変更する。更に、ステップ1305において、作業定義222を参照して、ステップ1303で作成した作業リストの内で作業の種別402が自動実行作業（「同期自動実行」及び「非同期自動実行」）である作業を決定し、そのノード名リストを作成する。

【0095】最後に、ステップ1301で受取った案件キーから次実行作業リスト212のレコードを特定し、作業リスト802に（ステップ1305で）作成した作業のノード名リストを追加する（ステップ1306）。

【0096】図18は本実施形態の「ExecuteWork」手続の処理手順を示すフローチャートである。「ExecuteWork」手続は、呼出しパラメータとして案件キー、作業のノード名、及びトランザクション記述子を受取る。処理手順は以下の様になる。

【0097】まず、最初のステップ1401において、呼出しパラメータとして案件キー、作業のノード名、及びトランザクション記述子を受け取る。次に、ステップ1402において、ステップ1401で受取った作業のノード名から、作業定義222のレコードを特定し、プログラム情報403を取得する。

【0098】次に、取得したプログラム情報403及びトランザクション識別子を渡して、外部プログラム呼出処理部203を呼出す（ステップ1403）。外部プログラム呼出処理部203は、プログラム情報403を基に、自動実行業務処理部112を呼出す。この時、トランザクション識別子をパラメータとして自動実行業務処理部112へ受け渡す事で、業務処理とワークフロー実行制御処理部111の処理、及び前後に実行された作業に対する業務処理とを、同一のトランザクションで実行

する事が可能となる。

【0099】更に、ステップ1404において、作業定義222を基に作業の種別を取得し、「同期自動実行」か否かをチェックする。「同期自動実行」の場合、ステップ1405において、案件実行状態制御処理部204の「CompleteWork」手続を呼出し、作業の完了処理を行う。

【0100】最後に、図19、図20を用いて、トランザクション実行単位制御処理部202の処理手順を説明する。トランザクション実行単位制御処理部202は、「CompleteCurrentTransaction」手続、及び「ExecuteSucceedingTransaction」手続を有している。

【0101】「CompleteCurrentTransaction」手続は、作業の完了に伴って呼出され、完了した作業に引続いて実行される作業の中で、完了した作業と同一のトランザクション内で実行すべき自動実行作業（「同期自動実行」及び「非同期自動実行」作業）を決定し、夫々の作業の自動実行処理を案件実行状態制御処理部204の「ExecuteWork」手続を呼出す事で実行する。

【0102】「ExecuteSucceedingTransaction」手続は、作業の完了や案件投入によって発生した引続いて実行すべき自動実行作業の中で、別トランザクションに分けて行う処理を、新規にトランザクションを開始する事で実行する。

【0103】図19は本実施形態の「CompleteCurrentTransaction」手続の処理手順を示すフローチャートである。「CompleteCurrentTransaction」手続は、呼出しパラメータとして、案件キー、（完了した）作業のノード名、及びトランザクション記述子を受取る。処理手順は以下の様になる。

【0104】まず、最初のステップ1501において、呼出しパラメータとして、案件キー、作業のノード名、及びトランザクション記述子を受取る。次に、ステップ1502において、ステップ1501で受取った案件キーを用いて、次実行作業リスト212のレコードを特定し、作業リスト802を取得する。

【0105】次に、ステップ1503において、ステップ1501で受取った作業のノード名を実行単位作業リスト602内に含むレコードをトランザクション実行単位定義224において検索する。検索の結果得られたトランザクション実行単位定義224のレコードの実行単位作業リスト602、及びステップ1502で取得した作業リスト、の双方に存在する作業のノード名を選択する。

【0106】ステップ1503において、該当する作業が存在した場合（ステップ1504）、ステップ1505を実行する。該当する作業が存在しなかった場合、

10

20

30

40

50

「CompleteCurrentTransaction」手続の処理を完了する。

【0107】次に、ステップ1505において、ステップ1503で選択した作業を作業リスト802から削除する。更に、ステップ1506において、案件キー、選択した作業のノード名、及びトランザクション記述子をパラメータとして、案件実行状態制御処理部204の手続「ExecuteWork」を呼出し、自動実行作業の処理を行う。

【0108】ステップ1502からステップ1506は、ステップ1504において該当する作業が無くなるまで（同一トランザクションで処理すべき自動実行作業が存在しなくなるまで）繰り返される。

【0109】図20は本実施形態の「ExecuteSucceedingTransaction」手続の処理手順を示すフローチャートである。「ExecuteSucceedingTransaction」手続は、呼出しパラメータとして、案件キーを受取る。処理手順は以下のようになる。

【0110】まず、最初のステップ1601において、呼出パラメータとして案件キーを受取る。次に、ステップ1602において、トランザクションマネージャ114に対しトランザクションの開始を要求し、新規に開始されたトランザクションのトランザクション記述子を受取る。受取ったトランザクション記述子は、この手続の現在の処理におけるトランザクションとして設定する。

【0111】次に、ステップ1603において、ステップ1601で受取った案件キーを用いて次実行作業リスト212を検索し、作業リスト802を取得する。取得した作業リスト802が空ならば（ステップ1604）、現在のトランザクションのコミットをトランザクションマネージャ114へ要求し、「ExecuteSucceedingTransaction」手続を完了する。取得した作業リスト802が空でなければ（ステップ1604）、リストから任意の作業を1つ選択し、作業リスト802から削除する（ステップ1605）。

【0112】次に、案件キー、選択した作業のノード名、現在の処理トランザクションのトランザクション記述子をパラメータとして、案件実行状態制御処理部204の手続「ExecuteWork」を呼出し、選択した自動実行作業に対する処理を行う（ステップ1606）。更に、案件キー、選択した作業のノード名、現在の処理トランザクションのトランザクション記述子をパラメータとして、トランザクション実行単位制御処理部202の手続「CompleteCurrentTransaction」を呼出す。

【0113】これにより、ステップ1606で実行した自動実行作業と同一のトランザクションで処理すべき、他の自動実行作業に対する処理を行う事が出来る。最後

に、ステップ1608において、トランザクションマネージャ114に対し、現在の処理トランザクションのコミット要求を出す。

【0114】ステップ1602からステップ1608までの処理は、ステップ1604において作業リスト802が空になるまで繰り返される。以上が、ワークフロー実行制御処理部111の各構成要素の詳細な説明である。

【0115】以下、図7～図8のプロセス定義の例を用いて、ワークフロー実行制御処理部111の各構成要素が実際にどの様に動作するかを説明する。まず、クライアント業務処理部113は、クライアント要求受付処理部201の「Start」手続を呼出し、案件投入処理を行う。ここでは、案件キーとして「0001」を、トランザクション記述子として「NULL」を渡したとする。クライアント要求受付処理部201は、ステップ1003において新規にトランザクションを開始し、ステップ1004において案件実行状態制御処理部204の「StartProcess」手続を呼出す。

【0116】案件実行状態制御処理部204の「StartProcess」手続では、ステップ1202において案件実行状態211の初期化処理を行う。図5の作業定義222内には、ノード名「A」～「E」の5つのレコードが、図6の制御ノード定義223には、ノード名「x」、「y」、「s」、「e」の4つのレコードが存在する為、案件実行状態211のテーブルには、案件キー701の値が「0001」で、ノード名702の値が夫々「A」、「B」、「C」、「D」、「E」、「x」、「y」、「s」、「e」の9つのレコードが追加される。更に、ノード名702の値が、「A」～「E」の5つのレコードにおいて状態703は「Initial」に、「y」のレコードにおいて状態703「0」に初期化される。

【0117】次に、ステップ1203及びステップ1204において、「Start」制御ノード（ノード名「s」）から遷移する作業の状態703を「Ready」に変更する。この例では、遷移元301の値が「s」であるフロー定義221のレコードは一つのみで、このレコードの遷移先302の値は「A」（これは作業のノード名）である為、案件実行状態211において（案件キー「0001」で）ノード名「A」のレコードの状態703が「Ready」に変更される。作業「A」は自動実行作業ではない為、ステップ1206において、案件キー「0001」の作業リスト802を空に設定し、案件実行状態制御処理部204の「StartProcess」手続は完了する。

【0118】次に、ステップ1005において、上記の案件実行状態211に対する変更をコミットし、ステップ1006においてクライアント業務処理部113に対して完了が通知される。次実行作業リスト212の案件

キー「0001」のレコードにおいて、作業リスト802は空である為、ステップ1007で呼出されたトランザクション実行単位制御処理部202の「ExecuteSucceedingTransaction」手続を特に行うことなく終了する。

【0119】次に、「作業実行」作業「A」における業務処理から始まる一連の処理手順について説明する。まず、作業「A」の処理を行うユーザの操作するクライアント業務処理部113において、クライアント要求受付処理部201の「GetWorkList」手続を作業のノード名「A」をパラメータとして渡して呼出す。この時点では、案件キー701の値が「0001」で且つ作業のノード名が「A」のレコードの状態703の値は「Ready」となっている為、結果として受取る案件キーのリストには値「0001」が含まれる。

【0120】クライアント業務処理部113は、取得した案件キーのリストをユーザに表示する事で、リスト中から処理を行う案件をユーザに選択させる事が出来る。ユーザはクライアント業務処理部113を操作し、選択した案件のノード名「A」の作業に対する業務処理を行う。ここでは、ユーザは案件キー「0001」の案件を選択し、処理を行ったとする。

【0121】クライアント業務処理部113は、トランザクションマネージャ114に対し、新規トランザクションの開始を要求し、業務処理に対する業務データ123の更新をデータベース102に対して行った後、案件キー「0001」、作業ノード名「A」及びトランザクション記述子をパラメータにしてクライアント要求受付処理部201の「Complete」手続を呼出す。

【0122】クライアント要求受付処理部201では、受取ったパラメータを用いて、案件実行状態制御処理部204の手続「CompleteWork」を呼出す（ステップ1104）。

【0123】「CompleteWork」手続では、まず、案件キー「0001」で且つ作業ノード名が「A」の案件実行状態211のレコードを検索し、状態703の値を「Completed」へ更新する。更に、ステップ1303及びステップ1304において、作業「A」から遷移する作業「B」を特定し、作業「B」に対する案件実行状態211のレコードの状態703を「Ready」へ更新する。作業「B」は「同期自動実行」作業である為、「B」のみから成る作業リストが、次実行作業リスト212の案件キー「0001」で特定されるレコードの作業リスト802へ追加される（ステップ1305、1306）。この時点で作業リスト802は、ノード名「B」のみを含んでいる。

【0124】クライアント要求受付処理部201は、次に、トランザクション実行単位制御処理部202の手続「CompleteCurrentTransaction」を呼出す（ステップ1105）。この時、パラメ

ータとして、案件キー「0001」、作業ノード名「A」、トランザクション記述子が渡される。

【0125】トランザクション実行単位制御処理部202は、作業リスト802としてノード名「B」のみを含むリストを取得する（ステップ1502）。更に、ステップ1503において、ノード名「A」及び「B」の双方を含むトランザクション実行単位（実行単位ID601「1」、実行単位作業リスト602「A」、
「B」、「C」）をトランザクション実行単位定義224に見つけ、「B」を選択する。ステップ1505において、「B」を作業リスト802から削除する。これにより、案件キー「0001」の作業リスト802は空となる。次に、ステップ1506において、ノード名「B」の自動実行作業の処理を、案件実行状態制御処理部204の「ExecuteWork」手続を呼出す事で実行する。

【0126】案件実行状態制御処理部204は、ステップ1402において作業「B」のプログラム情報403を作業定義222より読み出し、「Program1」を取得する。更に、ステップ1403において、「Program1」の自動実行業務処理部112を呼出す。この時、自動実行業務処理部112に渡されるトランザクション記述子は、作業「A」の完了通知時にクライアント業務処理部113から渡されたものである。これにより、作業「A」における業務処理と、作業「B」における業務処理は、同一トランザクションで実行される事になる。作業「B」は「同期自動実行」である為、ステップ1405において案件実行状態制御処理部204は、「CompleteWork」手続を呼出し作業「B」の完了処理を行う。

【0127】「CompleteWork」手続では、案件キー「0001」の作業「B」の状態703を「Completed」へ更新し（ステップ1302）、作業「B」から遷移する次に実行すべき作業「C」を特定し（ステップ1303）、案件キー「0001」の作業「C」の状態703を「Ready」へ更新する（ステップ1304）。作業「C」は、自動実行作業である為、次実行作業リスト212の案件キー「0001」で特定されるレコードにおいて、作業リスト802へノード名「C」を追加する。この時点で作業リスト802はノード名「C」のみを含んでいる。

【0128】ここで、トランザクション実行単位制御処理部202の手続「CompleteCurrentTransaction」のステップ1506が完了する。更に、「CompleteCurrentTransaction」手続では、ステップ1502において、作業「C」のみを含む作業リストを取得し、ステップ1503において作業「C」を選択する。なぜなら、呼出しパラメータとして与えられた作業のノード名「A」と、作業リスト802内のノード「C」との双方

を含むトランザクション実行単位（実行単位ID601「1」、実行単位作業リスト602「A」、「B」、「C」）がトランザクション実行単位定義224に存在する為である。作業「C」に対しても、作業「B」と同様に、作業リスト802から削除し（ステップ1505）、案件実行状態制御処理部204の「ExecuteWork」手続を呼出す（ステップ1506）。

【0129】案件実行状態制御処理部204の「ExecuteWork」手続では、作業「C」に対するプログラム情報「Program2」を取得し（ステップ1402）、更に、ステップ1403において、「Program2」の自動実行業務処理部112を呼出す。この時も、自動実行業務処理部112に渡されるトランザクション記述子は、作業「A」の完了通知時にクライアント業務処理部113から渡されたものである。作業「C」は「非同期自動実行」である為、「ExecuteWork」手続はここで完了する。

【0130】再び、トランザクション実行単位制御処理部202の手続「CompleteCurrentTransaction」に戻り、ステップ1502において案件キー「0001」の空の作業リスト802を取得する。ここで、「CompleteCurrentTransaction」手続の処理が完了し、クライアント要求受付処理部201の「Complete」手続に処理が戻る。

【0131】クライアント要求受付処理部201の「Complete」手続では、ステップ1106において、現在の処理トランザクションのコミットをトランザクションマネージャ114へ要求する。このトランザクションは、クライアント業務処理部113が開始し、作業「A」の業務処理の実行に用いたものであり、その後、自動実行業務処理部112を呼出して作業「B」、「C」に対する業務処理を行うのに用いたものである。また、その間、ワークフロー実行制御処理部111が行ったワークフロー実行状態情報121に対する更新処理も、同一のトランザクション内で実行されている。これにより、作業「A」、作業「B」、作業「C」における業務処理は、同一トランザクションで実行される事になる。これは、ビジネスプロセス定義122のトランザクション実行単位定義224において指定した通りである。また、このトランザクション処理中に何らかのシステム障害が発生した場合、トランザクションはロールバックされ、作業「A」の業務処理の実行から全てやり直す事で障害回復が可能である。

【0132】作業「C」は「非同期自動実行」の為、その完了通知は（例えば自動実行業務処理部112の業務処理の延長で実行される）クライアント業務処理部113からクライアント要求受付処理部201の「Complete」手続を呼出す事で行われる。ここでも、クライアント業務処理部113からトランザクション記述子

を受取る事で、業務データ123に対する業務処理における更新と、ワークフロー実行制御処理部111の行う作業「C」の完了処理を同一トランザクションで行う事が出来る。以下、作業「C」の完了通知から始まるワークフロー実行制御処理部111の一連の動作について説明する。

【0133】クライアント業務処理部113は、業務処理に対する業務データ123の更新をデータベース102に対して行った後、案件キー「0001」、作業ノード名「C」及びトランザクション記述子をパラメータにしてクライアント要求受付処理部201の「Complete」手続を呼出す。クライアント要求受付処理部201では、受取ったパラメータを用いて、更に案件実行状態制御処理部204の手続「CompleteWork」を呼出す（ステップ1104）。

【0134】「CompleteWork」手続では、先ず、案件キー「0001」で且つ作業ノード名「C」の案件実行状態211のレコードを検索し、状態703の値を「Completed」へ更新する（ステップ1302）。更に、ステップ1303及びステップ1304において、作業「C」から遷移する作業「D」及び「E」を特定し、作業「D」及び「E」に対する案件実行状態211のレコードの状態703を「Ready」へ更新する。作業「D」及び「E」は共に「同期自動実行」作業である為、「D」「E」の2つの要素からなる作業リストが、次実行作業リスト212の案件キー「0001」のレコードの作業リスト802へ追加される（ステップ1305及びステップ1306）。この時点で作業リスト802は、ノード名「D」及び「E」のみを含んでいる。

【0135】クライアント要求受付処理部201は、次に、トランザクション実行単位制御処理部202の手続「CompleteCurrentTransaction」を呼出す（ステップ1105）。この時、パラメータとして、案件キー「0001」、作業ノード名「C」、トランザクション記述子が渡される。トランザクション実行単位制御処理部202は、作業リスト802としてノード名「D」及び「E」を含むリストを取得する（ステップ1502）。しかし、作業「C」及び「D」、若しくは作業「C」及び「E」を共に含む実行単位作業リスト602を持つレコードはトランザクション実行単位定義224には存在しない為（ステップ1503、ステップ1504）、「CompleteCurrentTransaction」手続を完了する。

【0136】クライアント要求受付処理部201の「Complete」手続は、次にステップ1106において、トランザクションマネージャ114に現在の処理トランザクションのコミットを要求する。これにより、作業「C」における業務処理と、作業「C」の完了処理が同一トランザクションとして実行される。更に、ステッ

ブ1108において、トランザクション実行単位制御処理部202の手続「ExecuteSucceedingTransaction」が呼出される。

【0137】「ExecuteSucceedingTransaction」手続では、ステップ1602において新規にトランザクションを開始し、ステップ1603において作業リスト802を取得する。この時点で、作業リスト802には、作業「D」及び「E」が格納されている。ステップ1605において、作業「D」を選択し（作業「E」を選択してもかまわない）、作業リスト802から削除する。この時点で、作業リスト802には作業「E」のみが含まれる。次に、作業「D」をパラメータにして、案件実行状態制御処理部204の手続「ExecuteWork」を呼出す。

【0138】「ExecuteWork」では、作業「B」の場合と同様にして、作業「D」に対する自動実行業務処理部「Program3」を呼出し、作業「D」に対する案件実行状態211のレコードの状態703を「Completed」へ更新する。

【0139】再び「ExecuteSucceedingTransaction」手続に処理が戻り、作業のノード名「D」をパラメータにして「CompleteCurrentTransaction」手続を呼出す（ステップ1607）。

【0140】「CompleteCurrentTransaction」手続では、ステップ1502において作業「E」のみからなる作業リスト802を取得する。更に、ステップ1503において作業「E」を選択する。なぜなら、呼出しパラメータとして与えられた作業のノード名「D」と、作業リスト802内のノード「E」との双方を含むトランザクション実行単位（実行単位ID601「2」、実行単位作業リスト「D」、作業「E」）がトランザクション実行単位定義224に存在する為である。ステップ1505において、作業「E」を作業リスト802から削除し、ステップ1506において、作業「E」をパラメータとして案件実行状態制御処理部204の手続「ExecuteWork」を呼出す。

【0141】案件実行状態制御処理部204の手続「ExecuteWork」では、作業「D」の場合と同様にして、作業「E」に対する自動実行業務処理部「Program4」を呼出し、作業「E」に対する案件実行状態211のレコードの状態703を「Completed」へ更新する。この時、自動実行業務処理部112に渡されるトランザクション記述子は、作業「D」において呼出された自動実行業務処理部112に渡したものと同一であり、作業「D」及び作業「E」に対する業務処理は共に同一トランザクションで実行される。

【0142】再び、「CompleteCurrentTransaction」手続に処理が戻り、ステップ

1502において空の作業リスト802を取得し、「CompleteCurrentTransaction」手続を完了する（ステップ1503、ステップ1504）。

【0143】ここで、「ExecuteSucceedingTransaction」手続に処理が戻り、ステップ1608において、現在の処理トランザクションをコミットする。この時点で、作業「D」及び作業「E」に対して呼出された自動実行業務処理部112の業務処理が、同一トランザクションとしてコミットされる。これは、ビジネスプロセス定義122のトランザクション実行単位定義224において指定した通りである。また、このトランザクション処理中に何らかのシステム障害が発生した場合、トランザクションはロールバックされ、次実行作業リスト212の案件キー「0001」のレコードの作業リスト802に、作業「D」及び「E」が含まれる状態に戻る。

【0144】これは、障害を起こしたトランザクションの開始前に次実行作業リスト212に格納されていた次に自動実行すべき作業のリストが、障害発生後も失われずに（案件毎に）次実行作業リスト212に保持されている事を示している。その為、ワークフロー実行制御処理部111は、システム障害回復時に、全ての案件に対して「ExecuteSucceedingTransaction」手続を呼出す事で、自動実行作業の処理を継続する事が出来る。

【0145】以上で、本実施形態において、プロセス定義122中に定義された全ての作業に対する業務処理が、トランザクション実行単位定義224に指定された通りのトランザクション実行単位で実行される事が示された。

【0146】以上説明した様に本実施形態のワークフロー管理装置によれば、ワークフローによって処理される複数の作用を同一トランザクションとして実行するので、ワークフローによって処理される複数の作業の信頼性を高めることが可能である。

【0147】また本実施形態のワークフロー管理装置によれば、同一のトランザクションとして実行する複数作業を決定する定義情報をプロセス定義内に含んでいるので、複数作業を同一のトランザクションで実行する際に、個々の業務プログラムを変更すること無く業務プログラムの組合せを柔軟に変更することが可能である。

【0148】また本実施形態のワークフロー管理装置によれば、業務プログラムの処理からトランザクション制御のロジックを切り離しているため、業務プログラムの部品としての独立性を高め再利用し易くすることが可能である。

【0149】また本実施形態のワークフロー管理装置によれば、複数作業の実行とそのワークフロー実行状態情報の更新処理とを同一のトランザクションとして実行す

10

20

30

40

50

るので、複数作業における複数業務プログラム処理とそのワークフローの実行管理の一貫性を高めることが可能である。

【0150】

【発明の効果】本発明によればワークフローによって処理される複数の作用を同一トランザクションとして実行するので、ワークフローによって処理される複数の作業の信頼性を高めることが可能である。

【図面の簡単な説明】

【図1】本実施形態のワークフロー管理システムの構成例を示す図である。

【図2】本実施形態の各計算機の概略構成を示す図である。

【図3】本実施形態の動作原理の概略を示す図である。

【図4】本実施形態のワークフロー管理装置101の概略構成を示す図である。

【図5】本実施形態の作業定義222の含む定義項目と具体的な定義情報の例を示す図である。

【図6】本実施形態の制御ノード定義223の含む定義項目と具体的な定義情報の例を示す図である。

【図7】本実施形態のフロー定義221の含む定義項目と具体的な定義情報の例を示す図である。

【図8】本実施形態のトランザクション実行単位定義224の含む定義項目と具体的な定義情報の例を示す図である。

【図9】本実施形態のプロセス定義ツール115の表示画面を示す図である。

【図10】本実施形態のプロセス定義ツール115の保持する定義情報をプロセス定義122として格納する処理の処理手順を示すフローチャートである。

【図11】本実施形態の案件実行状態211の項目を示す図である。

【図12】本実施形態の次実行作業リスト212の項目を示す図である。

【図13】本実施形態の「Start」手続の処理手順を示すフローチャートである。

【図14】本実施形態の「Complete」手続の処理手順を示すフローチャートである。

【図15】本実施形態の「StartProcess」手続の処理手順を示すフローチャートである。

【図16】本実施形態の「TransitionSol

ver」手続の処理手順を示すフローチャートである。

【図17】本実施形態の「CompleteWork」手続の処理手順を示すフローチャートである。

【図18】本実施形態の「ExecuteWork」手続の処理手順を示すフローチャートである。

【図19】本実施形態の「CompleteCurrentTransaction」手続の処理手順を示すフローチャートである。

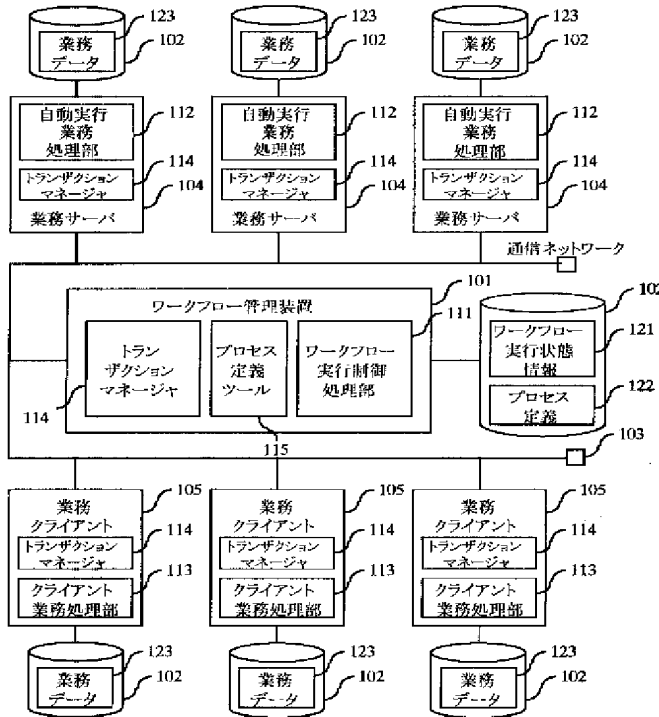
【図20】本実施形態の「ExecuteSucceedingTransaction」手続の処理手順を示すフローチャートである。

【符号の説明】

101…ワークフロー管理装置、102…データベース、103…通信ネットワーク、104…業務サーバ、105…業務クライアント、111…ワークフロー実行制御処理部、112…自動実行業務処理部、113…クライアント業務処理部、114…トランザクションマネージャ、115…プロセス定義ツール、121…ワークフロー実行状態情報、122…プロセス定義、123…業務データ、3001…表示装置、3002…CPU、3003…通信装置、3004…キーボード、3005…マウス、3006…メモリ、3007…補助記憶装置、4001…業務プログラム呼出し、4002…更新処理、4003…トランザクション開始要求、4004…トランザクションコミット処理要求、201…クライアント要求受付処理部、202…トランザクション実行単位制御処理部、203…外部プログラム呼出処理部、204…案件実行状態制御処理部、211…案件実行状態、212…次実行作業リスト、221…フロー定義、222…作業定義、223…制御ノード定義、224…トランザクション実行単位定義、401…ノード名、402…種別、403…プログラム情報、501…ノード名、502…種別、301…遷移元、302…遷移先、601…実行単位ID、602…実行単位作業リスト、2001…レイアウトボックス、2002…ツールボックス、2003…定義属性設定ダイアログボックス、2004…マウスポインタ、2011～2017…定義部品、2021…アロー、2022…作業、2023～2026…制御ノード、2027…トランザクション実行単位、701…案件キー、702…ノード名、703…状態、801…案件キー、802…作業リスト。

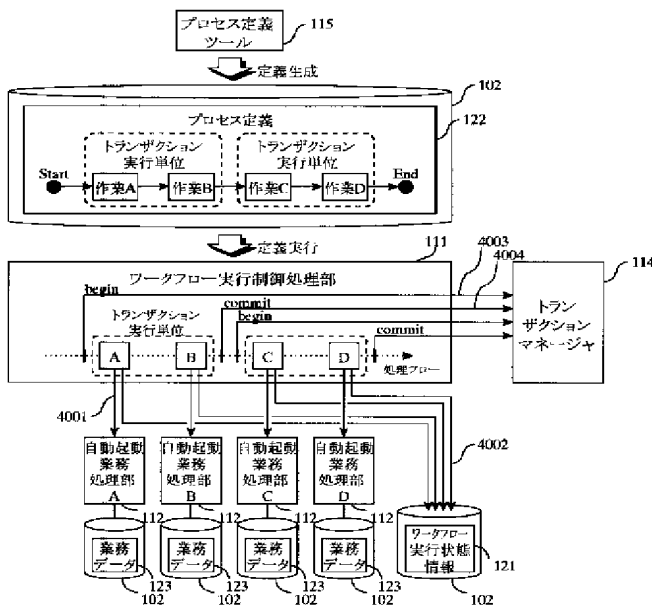
【図1】

図1



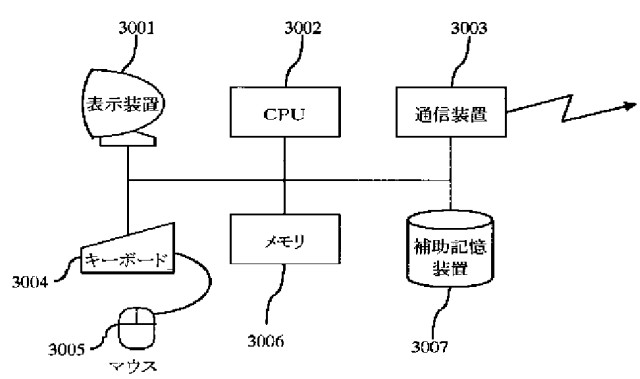
【図3】

図3



【図2】

図2



【図5】

図5

作業定義		
ノード名	種別	プログラム情報
A	作業者実行	-
B	同期自動実行	Program1
C	非同期自動実行	Program2
D	同期自動実行	Program3
E	同期自動実行	Program4

【図6】

図6

制御ノード定義	
ノード名	種別
x	Split
y	Join
s	Start
e	End

【図8】

図8

トランザクション実行単位定義	
実行単位 ID	実行単位 作業リスト
1	A, B, C
2	D, E

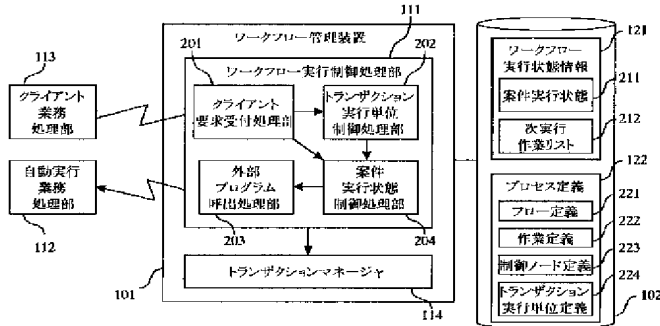
【図7】

図7

フロー定義	
遷移元	遷移先
s	A
A	B
B	C
C	x
x	D
x	E
D	y
E	y
y	e

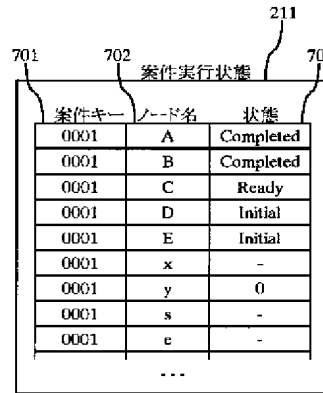
【図4】

図 4



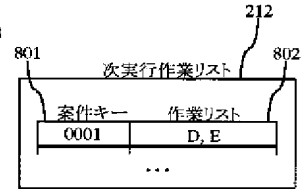
【図11】

図 1 1



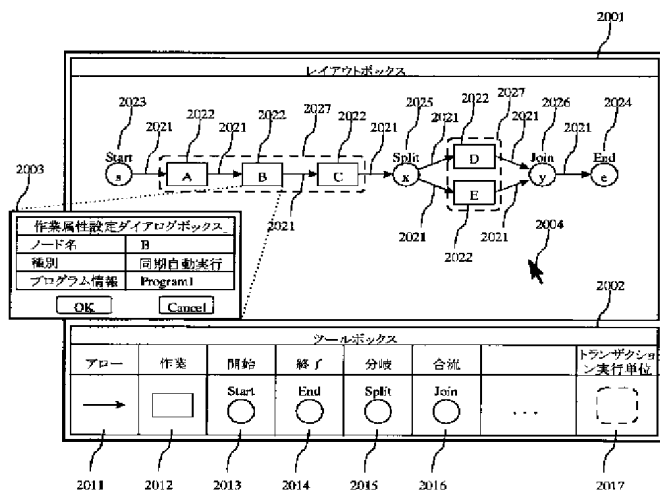
【図12】

図 1 2



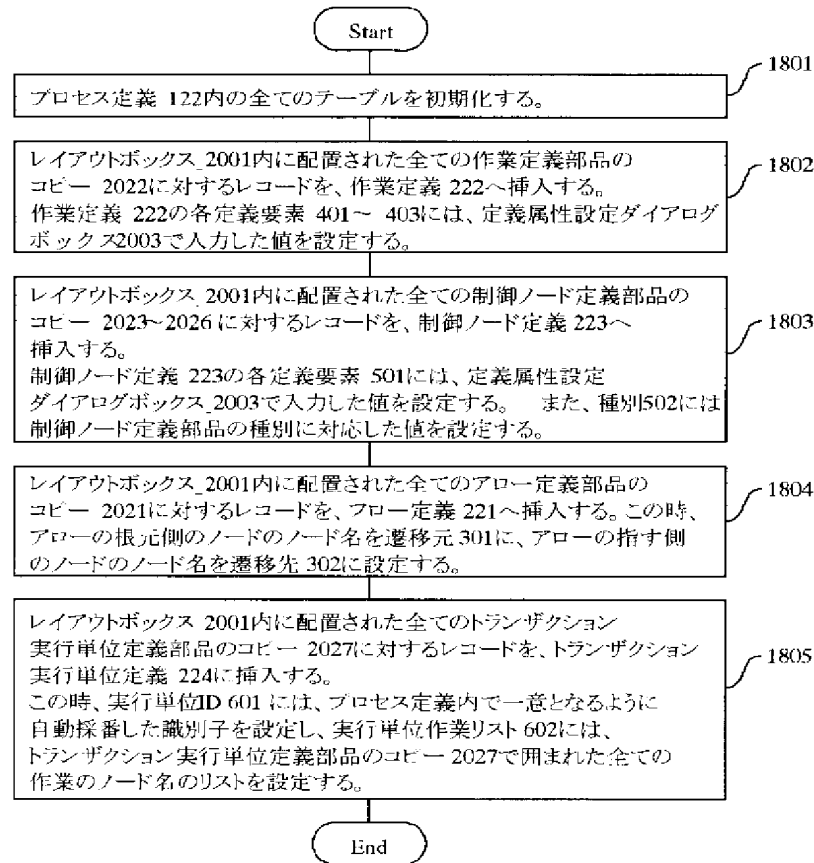
【図9】

図 9



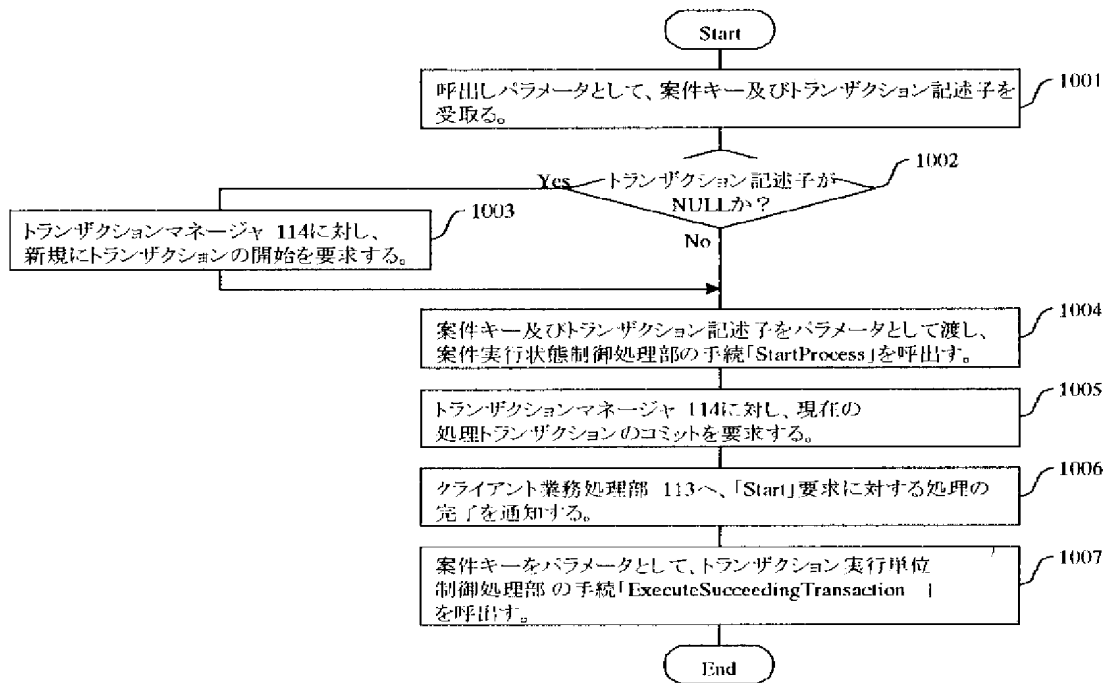
【図10】

図 10



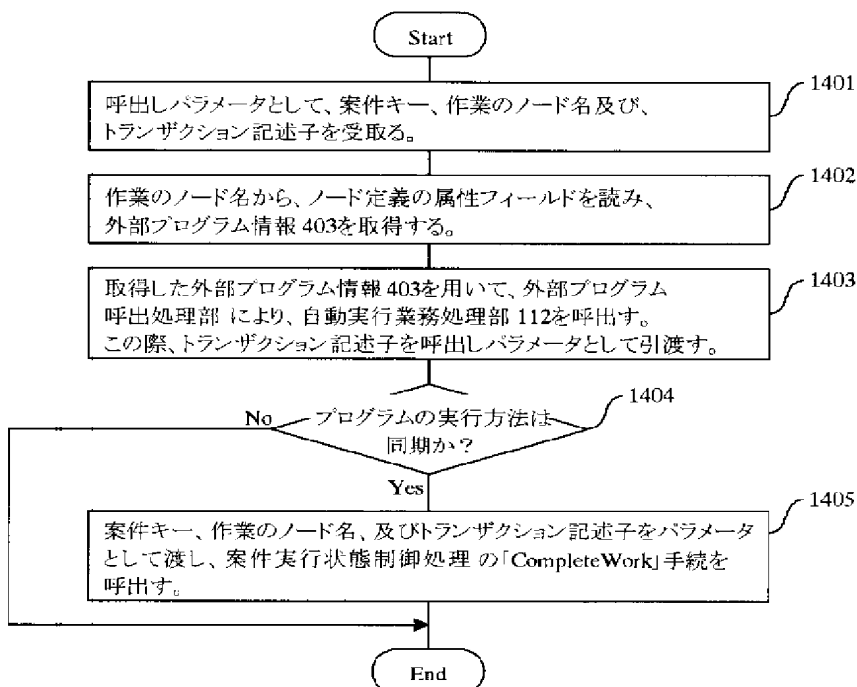
【図 1 3】

図 1 3



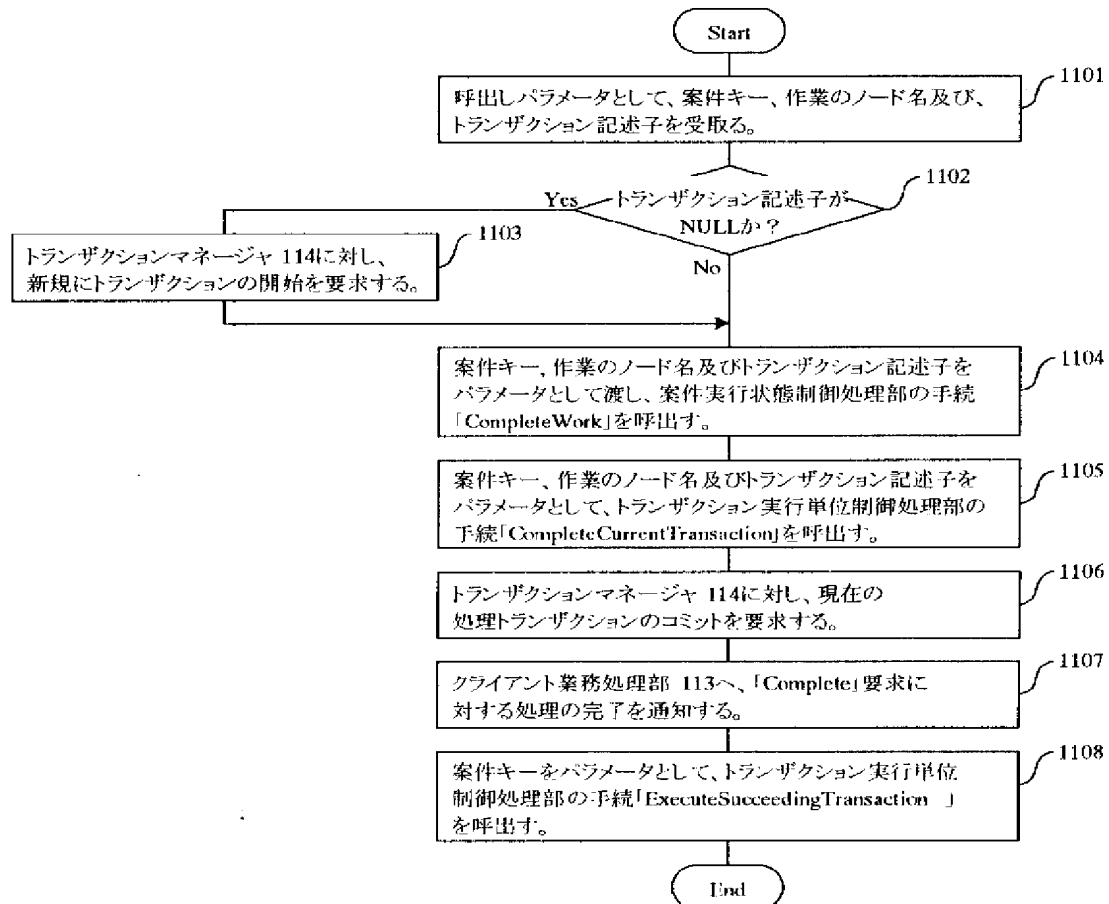
【図 1 8】

図 1 8



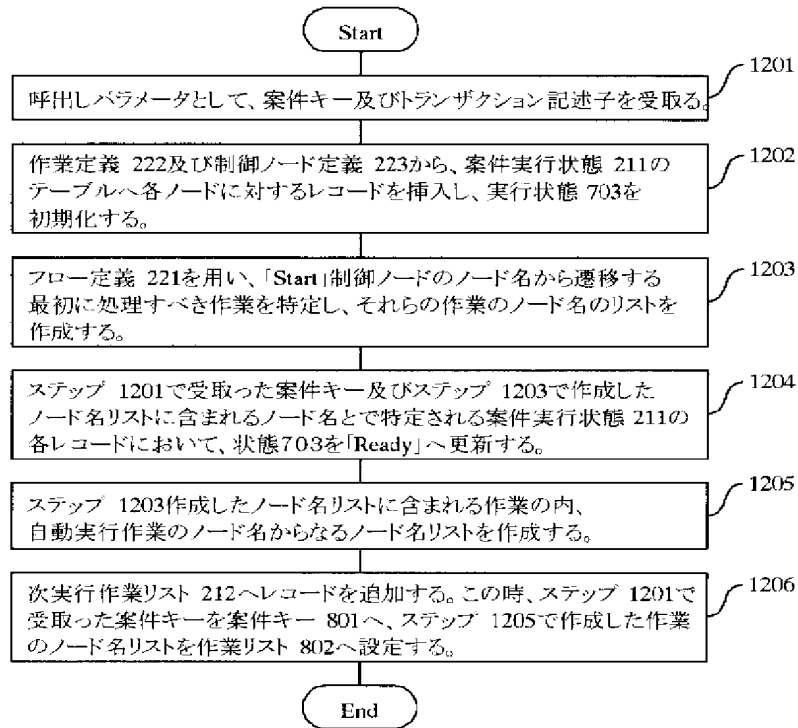
【図14】

図 14



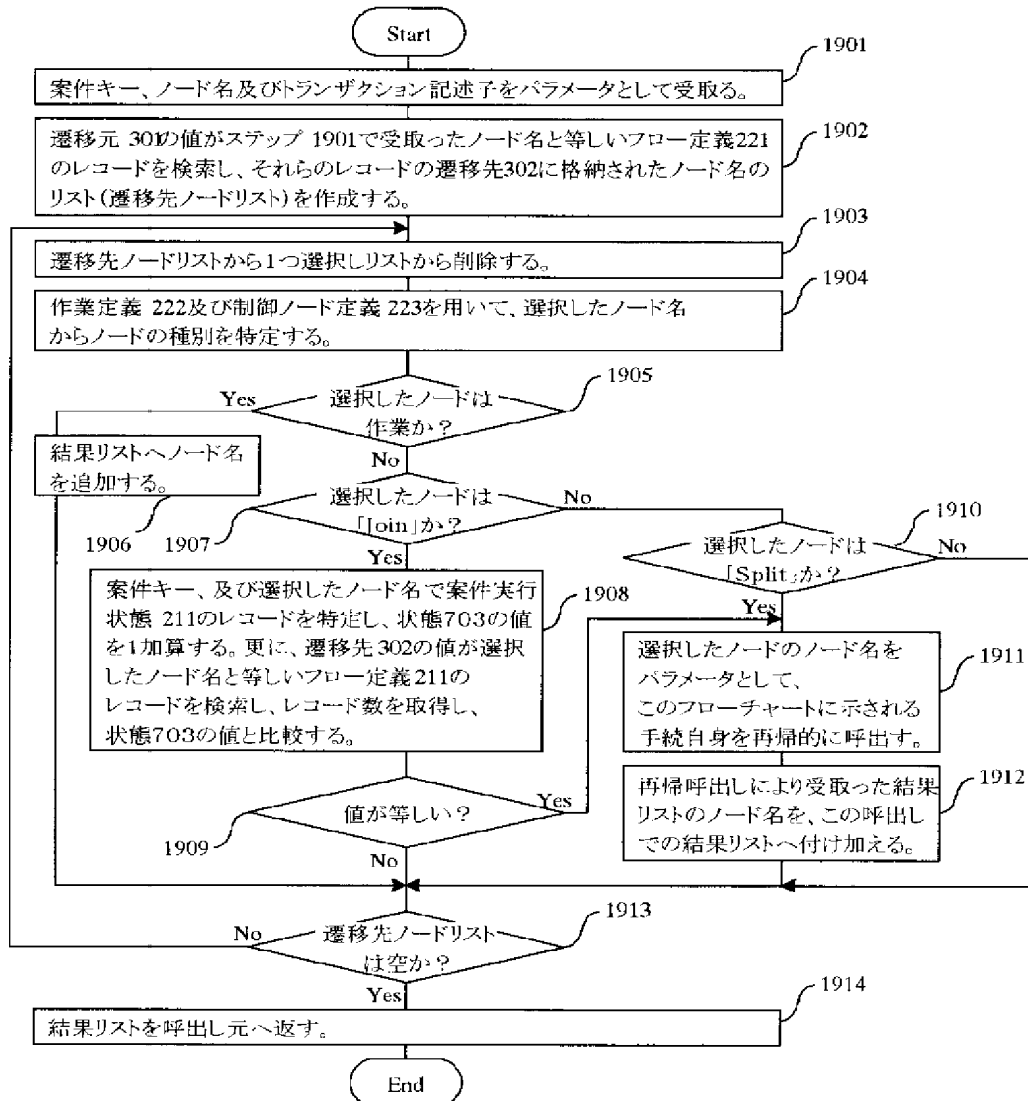
【図 1 5】

図 1 5



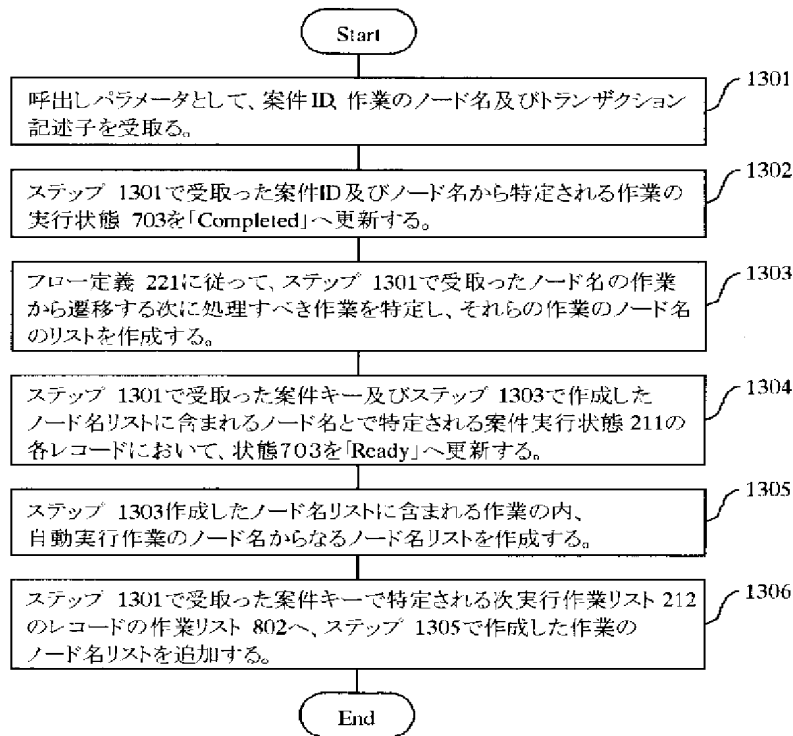
【図16】

図 1 6



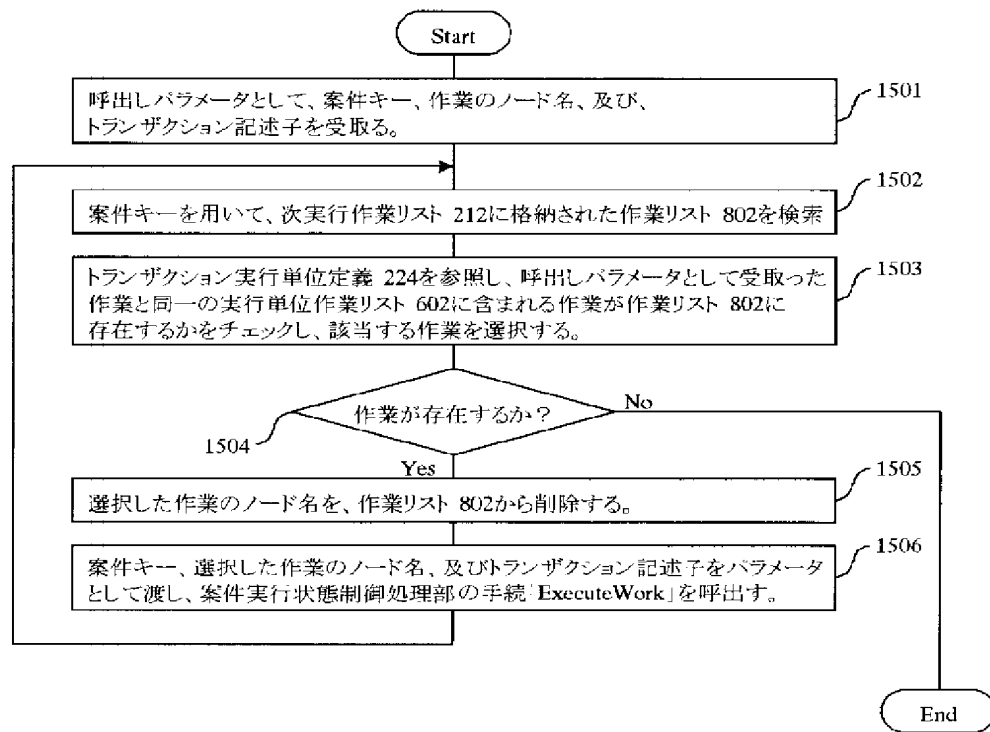
【図17】

図17



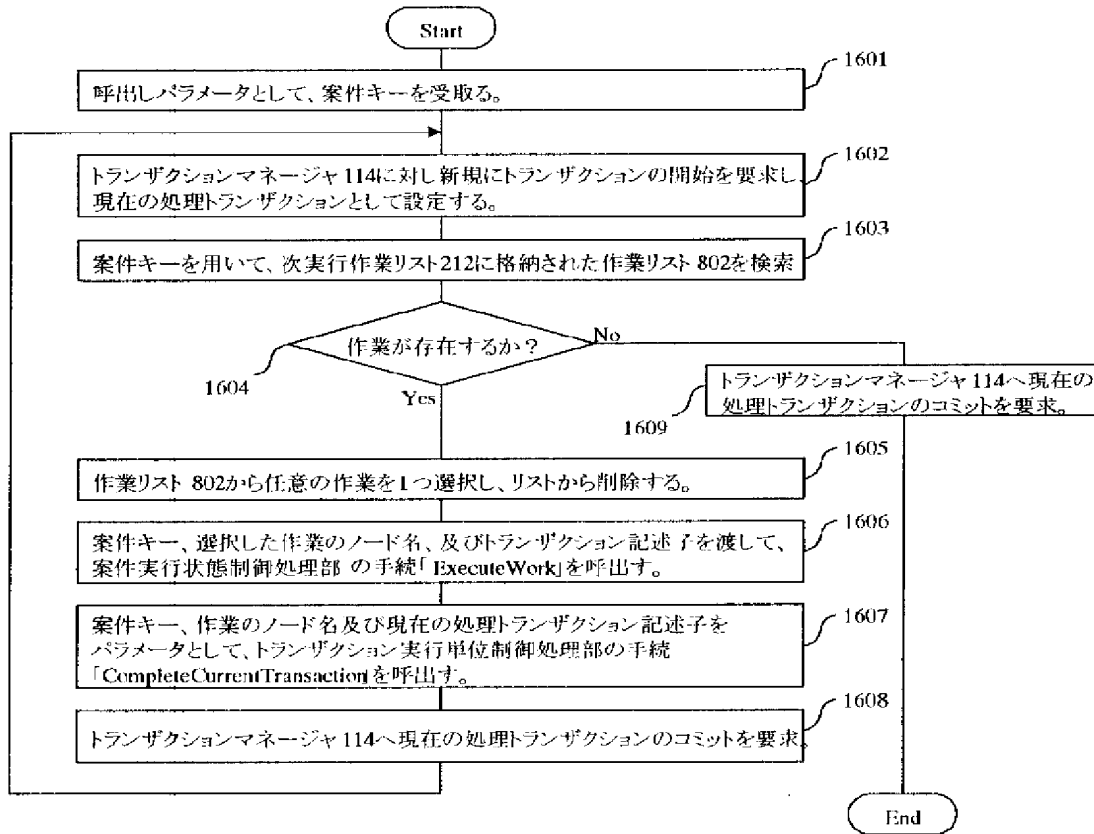
【図19】

図 19



【図20】

図 20



フロントページの続き

(72)発明者 田坂 光伸
神奈川県川崎市幸区鹿島田890番地 株式
会社日立製作所システム開発本部内

(72)発明者 村瀬 彰一
神奈川県川崎市幸区鹿島田890番地 株式
会社日立製作所システム開発本部内

F ターム(参考) 5B049 AA06 BB07 CC21 CC32 EE31
FF03 GG02 GG04
5B082 AA05 GB06
5B098 AA10 GC00 GC16

DERWENT-ACC-NO: 2001-130607

DERWENT-WEEK: 200114

COPYRIGHT 2008 DERWENT INFORMATION LTD

TITLE: Workpiece flow management for
computer, involves controlling and
managing progress condition operation
by determining transaction in operating
unit to start transaction

INVENTOR: KUREYAMA N; MIZOTE Y ; MURASE S ;
TASAKA M

PATENT-ASSIGNEE: HITACHI LTD[HITA]

PRIORITY-DATA: 1999JP-153330 (June 1, 1999)

PATENT-FAMILY:

PUB-NO	PUB-DATE	LANGUAGE
JP 2000348111 A	December 15, 2000	JA

APPLICATION-DATA:

PUB-NO	APPL-DESCRIPTOR	APPL-NO	APPL-DATE
JP2000348111A	N/A	1999JP-153330	June 1, 1999

INT-CL-CURRENT:

TYPE	IPC DATE
CIPP	G06F12/00 20060101
CIPS	G06F19/00 20060101
CIPS	G06F9/46 20060101
CIPS	G06Q10/00 20060101

ABSTRACTED-PUB-NO: JP 2000348111 A

BASIC-ABSTRACT:

NOVELTY - The operation of the progress condition is controlled and managed by determining the transaction in an operating unit to start the transaction during a multiple operation process. A series of operating processes are started based on the received process demand.

DESCRIPTION - INDEPENDENT CLAIMS are also included for the following:

- (a) a workpiece flow management operating apparatus;
- (b) and a recording medium for a processing program.

USE - For computer.

ADVANTAGE - Enhances reliability of workpiece flow operating processes.

DESCRIPTION OF DRAWING(S) - The figure shows the block diagram of a workpiece management apparatus.

CHOSEN-DRAWING: Dwg.4/20

TITLE-TERMS: WORKPIECE FLOW MANAGEMENT
COMPUTER CONTROL MANAGE
PROGRESS CONDITION OPERATE
DETERMINE TRANSACTION UNIT
START

DERWENT-CLASS: T01

EPI-CODES: T01-F02; T01-H; T01-J; T01-J05A;

SECONDARY-ACC-NO:

Non-CPI Secondary Accession Numbers: 2001-096727